02180: Introduction to Artificial Intelligence
Solutions to exercises at week 2
Missionaries and cannibals

The missionaries and cannibals problem is usually stated as follows. Three missionaries and three cannibals are on one side of a river, along with a boat that can hold one or two people. Find a way to get everyone to the other side without ever leaving a group of missionaries in one place outnumbered by the cannibals in that place. Whenever the boat goes from one side to the other, everybody has to go out before the boat can go back again. The boat can not sail without anybody in it.

1. A *state* is a description of all relevant parameters of a given problem (in this case it should include the positioning of all missionaries and cannibals with respect to the side of the river, and the position of the boat). How can the possible states of the problem be represented? What is then the initial state? Draw a diagram of the complete state space. What is the size of the state space (measured in number of states it contains)?

2. Why do you think people have a hard time solving this puzzle, given that the state space is so simple?

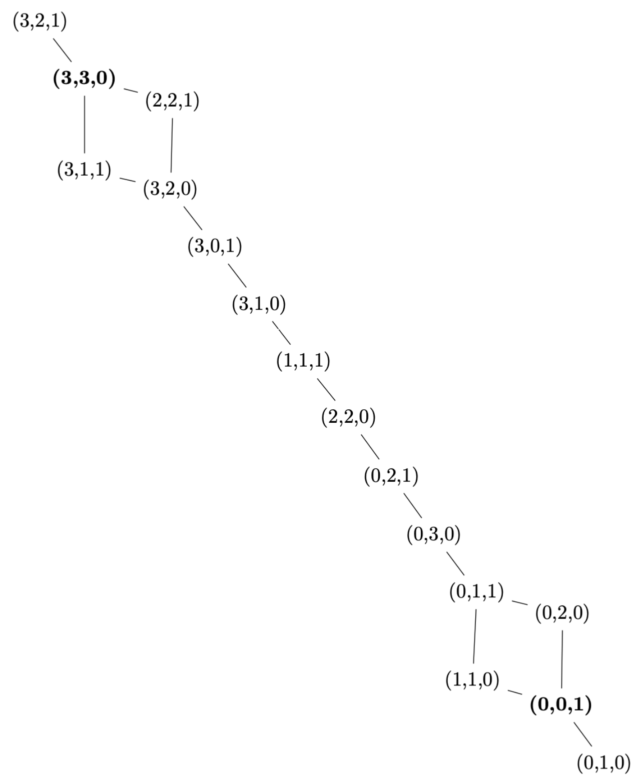
3. Assume instead that there are n missionaries and n cannibals. What is then the size of the state space? You can report the size of the full state space, including the states that are not reachable without violating the constraint of cannibals not outnumbering missionaries. You can measure the size of the state space in number of distinct states. Does the size of the state space imply that the problem is difficult for a computer to solve for big values of n?

4. Assume that in addition to the n missionaries, n cannibals there are now also n boats. Does this imply that the problem is difficult for a computer to solve for big values of n?

5. Assume that in addition to the n missionaries, n cannibals and n boats there are now also n banks instead of only 2. This means that any of the persons and boats can be at any of n banks. Does this imply that the problem is difficult for a computer to solve for big values of n?

Solution

1. In each state we only have to keep track of how many missionaries and cannibals are at each bank. Since the number of missionaries and cannibals at the right bank can be deduced from the number of missionaries and cannibals at the left, we only need to keep track of how many of each are at the left bank. The only additional thing we have to keep track of is whether the boat is at the left or right bank. Hence each state can be represented as a triple (m, c, b), where $m \in \{0, \ldots, 3\}$ is the number of missionaries at the left bank, $n \in \{0 \ldots 3\}$ is the number of cannibals at the left bank, and $b = 0$ if the boat is at the left bank, otherwise $b = 1$. Hence an upper bound on the size of the reachable part of the state space is $4 \times 4 \times 2 = 32$. The initial state can be assumed to be $(3, 3, 0)$ (everything initially at the left bank). The goal state is then $(0, 0, 1)$. The available actions are for the boat to take 1?2 people from the current bank to the opposite bank. However, we also have to make sure that the number of cannibals at any bank never outnumber the missionaries. This leads to the following state space (edge labels left out):



2. The are several reasons. One of them is that it is a bit tricky to keep track of which moves are legal. Another reason is that the solution is relatively long, 11 steps. Long solutions are not always a problem to find for humans, but it is only easy for us when we have a good heuristics to guide our search. For instance, finding a walking route from A to B on a map is usually not difficult for us, even if there are more than 11 intersections where we have to make a choice. Usually, it is sufficient in any intersection to choose the next road segment to be the one that ends up in a new intersection with minimal straight line distance to B (technically speaking, we would be doing a greedy best-first search where the heuristics is the straight line distance to B). However, for the missionary and cannibals problem, there is no simple heuristics to guide the search. Indeed, the obvious heuristics which counts the number of people not yet at the right bank is not particularly efficient, as in every second move, we have to move further away from the solution according to this heuristics (we need to move people back from the right to the left bank). Finally, for humans it is not so easy to keep track of which states have already been visited, so we easily end up in loops (a greedy best-first search with the mentioned heuristics would end up in an infinite loop unless it keeps track of the already visited states).

3. With n cannibals and n missionaries, the state can now be represented as an element of $[0, n] \times [0, n] \times \{0, 1\}$. There are $2(n+1)^2$ such triples, which is hence the size of the (full) state space. As there is hence still only a polynomial number of states in n, it should not be too hard for a computer to solve it even for large values of n. Actually, it has been proven that there is no solution to the problem with $n > 3$, but a computer searching for a solution might of course not realise that until having explored the full reachable state space.

4. Now states can be represented as elements $(m, c, b) \in [0, n] \times [0, n] \times [0, n]$, where m is the number of missionaries at the left bank, c is the number of cannibals at the left bank, and b is the number of boats at the left bank. The size of the state space becomes $(n+1)^3$, but this is still polynomial in n and should not be too hard for a computer.

Take the case $n = 5$. We can represent the missionaries as a list of dots:

$$\cdot\cdot\cdot\cdot\cdot$$

We can then place $n-1$ vertical bars to represent how they are distributed into the n banks. For instance:

$$\cdot|\cdot\cdot||\cdot\cdot|$$

means that there is 1 missionary on the first bank, 2 on the second, none on the third, 2 on the fourth, and none on the fifth. The number of distributions of missionaries is equal to the number of ways we can position the $n-1$ vertical bars. Each bar can be positioned in $n+1$ different ways. This gives in total $(n+1)^{(n-1)}$ different possibilities. But this is assuming that the bars are distinguishable, which they are not, so we have to divide by $n!$. So the number of possible distributions of missionaries is

$$\frac{(n+1)^{(n-1)}}{n!}.$$

We have the same number of possible distribution of cannibals and boats. This expression grows exponentially in n which can be seen by the following calculations:

$$\frac{(n+1)^{(n-1)}}{n!} \geq \frac{1}{n}\Pi_{i=1}^{n-1}\frac{n+1}{i} \geq \frac{1}{n}\Pi_{i=1}^{n-1}\frac{n}{i} \geq \Pi_{i=2}^{n-1}\frac{n}{i} = \Pi_{i=2}^{\lfloor n/2 \rfloor}\frac{n}{i} + \Pi_{i=\lfloor n/2 \rfloor+1}^{n-1}\frac{n}{i}$$
$$\geq \Pi_{i=2}^{\lfloor n/2 \rfloor}\frac{n}{i} \geq \Pi_{i=2}^{\lfloor n/2 \rfloor}\frac{n}{(n/2)} \geq \Pi_{i=2}^{\lfloor n/2 \rfloor}2 \geq 2^{\lfloor n/2 \rfloor-1} \geq 2^{(n/2)-2} = \frac{1}{4}(\sqrt{2})^n.$$

This means that in principle it might be very difficult for a computer to solve the problem for big values of n, unless it employs an efficient heuristics to avoid having to explore the entire state space. Having more banks available than in the version considered in question (e) ought to make the problem easier to solve, since it is easier to satisfy the constraint of never having the cannibals outnumber the missionaries at any bank. In fact, there is a trivial solution in this case: use half of the boats to bring all the cannibals to the goal bank, then use the rest to bring all the missionaries over there. But even though the problem should be easier in principle (and easier in practice for a human), it still might create more problems for a (naively implemented) computer program, since there are many more available actions to consider. For instance, a breadth-first search will certainly run out of time and memory for large values of n: it will explore an exponentially large state space (in n) before finding a solution.